

# Übungsblatt 2 zur Vorlesung Compilerbau

Abgabe: 10.11.2024, 23:59

Prof. Dr. Michael Kuhn (michael.kuhn@ovgu.de)

Michael Blesel (michael.blesel@ovgu.de)

Parallel Computing and I/O • Institut für Intelligente Kooperierende Systeme

Fakultät für Informatik • Otto-von-Guericke-Universität Magdeburg

<https://parcio.ovgu.de>

---

Auf diesem Übungsblatt werden wir das erlernte Wissen über Scanning benutzen, um einem Scanner für JSON zu implementieren

## 1. JSON-Scanner (300 Punkte)

Implementieren Sie mit Hilfe von Kapitel 3.7 des Compilerbooks und des Werkzeugs Flex einen Scanner für JSON. Die genauen Spezifikationen von JSON finden Sie unter <https://json.org/>. Genauere Informationen über die Syntax von Flex finden Sie unter <https://www.cs.princeton.edu/~appel/modern/c/software/flex/flex.html>.

In den Materialien zu diesem Übungsblatt finden Sie den Code für das Grundgerüst Ihres Scanners und ein zugehöriges Makefile. Im Unterverzeichnis json finden Sie außerdem zwei Beispiel-JSON-Dateien, welche Sie zum Testen der Funktionalität Ihrer Implementation verwenden können.

Ihr fertiger Scanner sollte zum Beispiel für `simple.json` die folgende Ausgabe produzieren.

```
TOKEN_OBJECT_BEGIN : {  
TOKEN_STRING : "fruit"  
TOKEN_COLON : :  
TOKEN_STRING : "Apple"  
TOKEN_COMMA : ,  
TOKEN_STRING : "size"  
TOKEN_COLON : :  
TOKEN_NUMBER : 1  
TOKEN_COMMA : ,  
TOKEN_OBJECT_END : }
```

Für die Bearbeitung dieser Aufgabe auf dem Cluster denken Sie daran den Softwarestack und das Flex Modul zu laden.

```
$ . /opt/spack/cb-202425/env.sh  
$ module load flex
```

Die folgenden Aufgaben sind zu lösen:

- Erweitern Sie den Scanner so, dass die komplette JSON-Syntax unterstützt wird.

- Erweitern Sie den Scanner so, dass Sie den Pfad zu einer beliebigen JSON-Datei als Argument angeben können und testen Sie die Funktionalität mit `test.json`.
- **Frage:** Testet `test.json` alle möglichen Formen der JSON-Syntax? Falls nein, geben Sie Beispiele. Unterstützt Ihre Implementation diese Beispiele?

## Abgabe

Als Abgabe werden wir den letzten Commit vor der Abgabefrist in Ihrem Git-Repository. Im Hauptverzeichnis des Repositories wird ein Verzeichnis `CB-2024-Uebung-02-Materialien` mit folgendem Inhalt erwartet:

- Eine Datei `gruppe.md` mit den Gruppenmitgliedern (eines je Zeile) im folgenden Format:  
Erika Musterfrau <erika.musterfrau@example.com>  
Max Mustermann <max.mustermann@example.com>
- Der überarbeitete Code des Scanner Programms (Aufgabe 1)
- Eine Datei `antworten.md` mit ihren Antworten (Aufgabe 1)